REMARKS

The comments of the applicant below are each preceded by related comments of the examiner (in small, bold type).

4. Claims 1-3,5, 7, 8, 10, 11, 13, 14, 16, 18, 19, 21, 22, 24-26, 39, 40, 42-44, 46-49, 52 and 53 are rejected under 35 U.S.C. 102(e) as being anticipated by Makphaibulchoke et al. (US PG Publication 2003/0014616 Al), hereinafter Makphaibulchoke.

As for claims 1,2, 7,13, 18, 47 and 48, Makphaibulchoke teaches a method (and computer program product) comprising:

maintaining a memory management table that includes one or more entries, each entry defining a location of a portion of data stored within a memory system (paragraphs 0019 and 0027, an ACPI table is used to store data structures; pointers are used to reference locations of the data structures);

from at least two types of endian conversion, including a first type to convert data to a first endian format, and a second type to convert data to the first endian format, determining a type (data structures are stored in ACPI tables in little-endian format (in the preferred embodiment). Makphaibulchoke teaches converting the data structures in the tables from little to big-endian format. This conversion from little to big is either accomplished via byte-swapping, or bit reversal (two types of conversion to convert to a single endian format - paragraph 0019, all lines. More specifically. Fig. 5 elements 540 and 545 illustrate byte-swapping and bit reversal modules respectively. Paragraph 0024, all lines teaches illustrates how reversing the order of the bytes can produce the endian conversion. Alternately, paragraphs 0025 and 0031, all lines illustrates how this same conversion can be accomplished through reversal via shifting the bits); and

writing an entry to a memory management table based on the determining (paragraphs 0024 and 0025, all lines disclose that the conversion rearranges the data structures stored in the ACPI table. In other words, each entry in the table is converted either via bit reversal or byte swapping, hence what's written in the table is contingent upon the type of conversion that took place).

The applicant disagrees. Claim 1 has been amended to recite that the entry written to the memory management table specifies "one of the at least two types of endian conversion", a feature that *Makphaibulchoke* did not disclose and would not have made obvious.

In *Makphaibulchoke*, if a current data structure is larger than one byte, "byte swapping module **540** swaps the current data structure to convert it to big endian format." (para. 0025). Further, "[i]f the test at **630** determines that the current data structure contains at least one bit field, bit reversal module **545** reverses the bits in the data structure at **635**." *Id.* Though *Makphaibulchoke* discloses performing a test prior to performing an endian conversion, *Makphaibulchoke* fails to disclose writing an entry based on this test that specifies one of at least two types of endian conversion.

Applicant : Peter J. Barry et al.
Serial No. : 10/665,976
Filed : September 19, 2003
Page : 11 of 13

Attorney's Docket No.: 10559-849001 / P16875

As for claim 24, Makphaibulchoke teaches a memory management table residing in computer memory comprising:
one or more table entries, with each table entry having a first field for defining the location of a portion of data stored within a memory system and a second field for defining a type determined from at least two types of endian conversion (paragraphs 0024-0025, all lines - the bit field is a single bit which specifies in byte swapping or bit reversal is appropriate for the data structure. Additionally note in paragraph 0027, all lines - pointers are used to reference locations of the data structures), including a first type to convert data to a first endian format and a second type to convert data to the first endian format (as per the rationale in the rejection claim 1, supra).

The applicant disagrees. In claim 24, each table entry has "a second field for defining a type determined from at least two types of endian conversion." As explained above, *Makphaibulchoke* did not to disclose or would not have made obvious writing an entry specifying one of at least two types of endian conversion.

Further, *Makphaibulchoke*'s bit field is not (as the examiner contends) "a single bit which specifies in (*sic*) byte swapping or bit reversal is appropriate for the data structure." *Makphaibulchoke*'s bit field is an example of a data structure that may be contained in an ACPI table. (para. 0024). *Makphaibulchoke* describes that a "[b]it reversal module **545** ... reverses the bit order of data structures, particularly bit fields, contained within data collections such as ACPI tables." *Id.* "If the test at **630** determines that the current data structure contains at least one bit field, bit reversal module **545** reverses the bits in the data structure at **635**." (para. 0025). Therefore, *Makphaibulchoke* discloses using a bit reversal module to reverse the bits in a bit field data structure, but does not disclose that a bit field itself specifies if byte swapping or bit reversal is appropriate for a data structure.

As for claims 39 and 43, Makphaibulchoke teaches a method (and product) comprising:
accessing a table entry of a memory management table, wherein the table entry is associated with a portion of data stored within a memory system and includes a conversion-type indicator (paragraphs 0024-0025, all lines - the bit field is a single bit which specifies in byte swapping or bit reversal is appropriate for the data structure); and
from at least two types of endian conversion, including a first type to convert data to a first endian format and a second type to convert data to the first endian format, determining a type based on the conversion-type indicator (as discussed in the rejection of claim 1 presented supra).

The applicant disagrees. Claim 39 recites that "a table entry of a memory management table ... includes a conversion-type indicator."

As explained above, *Makphaibulchoke* does not disclose that a bit field specifies if byte swapping or bit reversal is appropriate for a data structure. Therefore, *Makphaibulchoke* fails to disclose or make obvious "accessing a table entry ... [that] includes a conversion-type indicator" as recited in claim 39. Claim 43 is patentable for at least similar reasons.

Amended claims 7, 13, 18, 29, 34, and 47 are patentable for at least similar reasons as those described above.

All of the dependent claims are patentable for at least the reasons for which the claims on which they depend are patentable.

Canceled claims, if any, have been canceled without prejudice or disclaimer.

Any circumstance in which the applicant has (a) addressed certain comments of the examiner does not mean that the applicant concedes other comments of the examiner, (b) made arguments for the patentability of some claims does not mean that there are not other good reasons for patentability of those claims and other claims, or (c) amended or canceled a claim does not mean that the applicant concedes any of the examiner's positions with respect to that claim or other claims.

The fee in the amount of $120 for the Petition for Extension of Time fee is being paid concurrently herewith on the Electronic Filing System (EFS) by way of Deposit Account authorization. Please apply all charges or credits to Deposit Account No. 06-1050, referencing Attorney Docket No. 10559-849001.

Applicant : Peter J. Barry et al.
Serial No. : 10/665,976
Filed : September 19, 2003
Page : 13 of 13

Attorney's Docket No.: 10559-849001 / P16875

Respectfully submitted,

Date: September 10, 2007

/Whitney A. Fellberg/
Whitney A. Fellberg
Reg. No. 59,173

Fish & Richardson P.C.
225 Franklin Street
Boston, MA 02110
Telephone: (617) 542-5070
Facsimile: (617) 542-8906

21647443.doc